8П

УДК 62-50:15

Н.Б. Паклин
ГОУ ВПО «Рязанский филиал Московского государственного университета экономики, статистики и информатики», г. Рязань, Россия
paklin@newmail.ru

Отбор переменных в логистическую регрессию генетическим алгоритмом

В статье исследуются эффективные процедуры отбора переменных в бинарные классифицирующие модели на основе логистической регрессии. Для этого используется генетический алгоритм, причем в функцию фитнеса особи параметр штрафа за включение в модель новых переменных изменяется в зависимости от рассчитанного значения площади под ROC-кривой. Проведены эксперименты на модельных наборах данных и в задаче кредитного скоринга.

Введение

В ряде прикладных областей, таких, как медицина и кредитный скоринг, логистическая регрессия [1] неизменно остается популярным средством для построения бинарных классифицирующих моделей, даже несмотря на появление в последние два десяти-летия эффективных алгоритмов машинного обучения. Причины этого лежат в том, что математический аппарат логистической регрессии хорошо изучен, коэффициенты регрессии поддаются интерпретации, а при помощи *ROC*-анализа можно подобрать точку отсечения (*cut-off value*) так, чтобы модель обеспечивала заданный уровень чувствительности. Последнее особенно важно в медицинском скрининге, в котором нужно добиваться высокой (более 90 %) чувствительности диагностического теста.

В последние годы бурный рост розничного кредитования в банковском секторе в России и странах бывшего СНГ заставил банки применять эффективные методики оценки заемщиков, или скоринг. Основным математическим средством для построения так называемых скоринговых карт по-прежнему остается логистическая регрессия, хотя накоплен значительный мировой опыт использования для этих целей деревьев классификации и искусственных нейронных сетей [2]. Как и задача медицинской диагностики, прогнозирование кредитоспособности заемщика на основе накопленных статистических данных – кредитных историй – сводится к задаче бинарной классификации. Типичной является ситуация, когда приходится иметь дело с десятками переменных, и, соответственно, производить их отбор для построения модели. Включение в модель регрессии шумового признака, никак не связанного с восстанавливаемой зависимостью, может только ухудшить обобщающую способность модели. Выбор оптимального набора переменных путем перебора всех комбинаций приводит к *NP*-полной задаче. Поэтому на практике получили распространение статистические процедуры пошагового отбора переменных, которые позволяют снизить количество вычислений, но не обеспечивают нахождения оптимального набора входных переменных ввиду «жадных» стратегий. Поэтому представляется актуальной разработка «нежадных» методов отбора, основанных на случайном направленном поиске.

В работе [3] уже была предпринята попытка адаптации простого генетического алгоритма к отбору переменных в бинарную логистическую регрессию, которая проверялась на задаче предсказания инфаркта миокарда по набору из 43 симптомов больного.

Результаты экспериментов показали превосходство генетического алгоритма над традиционными статистическими процедурами, однако введенный авторами параметр в функцию приспособленности подобран эмпирически для конкретной обучающей выборки. В данной работе продолжается исследование проблемы эволюционного отбора переменных в логистическую регрессию, предлагается модифицированная функция приспособленности и проводится тестирование на синтетических наборах данных, в том числе из *UC Irvine Machine Learning Repository*, а также на реальных кредитных историях.

Анализ «жадных» стратегий отбора

На практике при отборе переменных в регрессионную модель приходится реализовывать два противоречивых требования:

– нужно использовать как можно больше входных переменных, содержащих *новую* информацию о выходной переменной;

– поскольку каждая новая переменная может ухудшить обобщающую способность модели, нужно стремиться, чтобы модель содержала как можно меньше входных переменных.

Поиск наилучшей регрессионной модели, как правило, заключается в поиске компромисса между данными требованиями. Прикладная статистика предлагает две основные процедуры для отбора переменных: метод прямого выбора (англ.: *forward selection*) и метод обратного исключения (англ.: *backward elimination*). Для анализа недостатков рассмотрим их подробнее [1].

Процедура *forward* начинается с «пустой» модели, в которую еще не включена ни одна переменная. Она содержит следующие шаги.

1. Для первой переменной, вводимой в модель, основным критерием выбора является высокая корреляция с выходной переменной. Если полученная в результате модель не обладает достаточной значимостью, из этого следует, что среди доступных переменных исходной выборки значимые переменные отсутствуют. В противном случае переходят к шагу 2.

2. Для каждой из остальных переменных вычисляется последовательная F-статистика для данной переменной и переменных, уже включенных в модель. При этом каждый раз выбирается та переменная, для которой значение последовательной F-статистики будет наибольшим (обозначим ее $F_{\max}$).

3. Для значения $F_{\max}$ проводится тест значимости. Если модель после добавления переменной, выбранной на шаге 2, не обладает достаточной значимостью, то алгоритм останавливается и текущая модель остается без переменной, выбранной на шаге 2. В противном случае изменение модели принимается и осуществляется переход на шаг 2 для выбора следующей переменной.

Процесс продолжается до тех пор, пока все значимые переменные не будут включены в модель.

В отличие от метода *forward*, процедура *backward* начинается с «полной» модели (или модель *enter*), когда в нее включаются все доступные переменные. Процедура также содержит три шага.

1. Решается задача регрессии с помощью полной модели, т.е. когда в ней присутствуют все доступные переменные.

2. Для каждой переменной в модели вычисляется частная F-статистика. Предпочтение отдается переменной, для которой значение частной F-статистики будет наименьшим (обозначим его $F_{\min}$).

3. Производится тест значимости $F_{\min}$. Если $F_{\min}$ не является достаточно значимой, то связанная с ней переменная исключается из модели и производится возврат ко 2-му шагу. Если $F_{\min}$ имеет высокую значимость, то алгоритм останавливается, и формируется отчет о текущем состоянии модели. Если это первый проход алгоритма, то мы имеем полную модель и, следовательно, все доступные переменные являются значимыми. Если проход не является первым, то модель уменьшается на одну или несколько переменных.

Эти процедуры, по сути, являются алгоритмами оптимизации на большом пространстве наблюдений. По этой причине отсутствует гарантия, что действительно будет найдена наилучшая модель из всех возможных (глобальный оптимум), т.е. будет построена модель, обеспечивающая минимальную ошибку и максимальную значимость. Единственным способом гарантировать, что будет найдена действительно наилучшая модель из всех возможных, является перебор всех возможных комбинаций входных переменных, т.е. метод глобального поиска. Метод глобального поиска не применим на практике (требуется перебрать 2^k комбинаций, где k – число потенциальных переменных). Еще одна проблема – переобучение. В машинном обучении принято оценивать качество модели не только по ошибке классификации на обучающем множестве, но и по ошибке обобщения, которая рассчитывается на тестовом множестве. Кроме того, для бинарных классификаторов, в том числе логистической регрессии, применяется *ROC*-анализ [4], в котором анализируется индекс *AUC* – площадь под *ROC*-кривой. Эта кривая есть график зависимости чувствительности от специфичности, рассчитываемых при различных значениях точек отсечения. Значение *AUC*, рассчитываемое на обучающей и тестовой выборках, определяет прогностическую силу модели. $AUC = 0{,}5$ соответствует бесполезному классификатору, а $AUC = 1$ – идеальному. Считается, что регрессионная модель, имеющая высокое значение площади под кривой на обучающем множестве и низкое на тестовом, демонстрирует эффект переобучения. Рассмотренные статистические процедуры *forward* и *backward* никак не контролируют эффект переобучения, поскольку не используют в своей работе обращение к отдельному множеству, которое принято называть валидационным. Использование генетического алгоритма устраняет данный недостаток.

Формализация задачи отбора генетическим алгоритмом

Рассмотренные выше утверждения позволяют сформулировать целевую функцию для решения задачи отбора переменных в регрессионную модель: максимизация площади под кривой на валидационном (т.е. не участвующим в расчете коэффициентов регрессии) множестве и минимизация количества переменных. В терминах генетического алгоритма функция приспособленности будет выглядеть следующим образом [3]:

$$F(u,C,S) = AUC_S(m_C(u)) + \rho\frac{u-n}{u} \to \max\,, \qquad 1)$$

где C и S – обучающее и валидационное множества соответственно; n – число переменных, отобранных в модель (константа всегда включена в модель); u – общее число переменных; $m_C(u)$ – модель логистической регрессии, построенная на множестве C; $AUC_S(m_C(u))$ – площадь под *ROC*-кривой, рассчитанная на множестве S; ρ – параметр.

Первая часть функции (1) изменяется от 0,5 до 1. Выражение $(u-n)/u$ изменяется от 0 до 1. Параметр ρ регулирует соотношение между числом переменных в

модели и ее прогностической силой: чем он больше, тем меньше переменных будет в «лучшей» особи генетического алгоритма. Каждый ген особи и определяет, включать переменную в регрессионную модель или нет.

В [3] параметр ρ подобран эмпирически и установлен равным 0,02. Пробные эксперименты показали, что такое значение не является универсальным. При низких значениях AUC важно не ограничивать пространство поиска и как можно меньше штрафовать особь за увеличение количества переменных в модели. С повышением AUC нужно стремиться к снижению числа переменных, т.е. увеличивать штраф за ее добавление. Фиксированное значение ρ такую гибкость не обеспечивает.

Исходя из вышесказанного, предлагается следующая кусочно-линейная функция для ρ, зависящая от AUC_S (рис. 1).

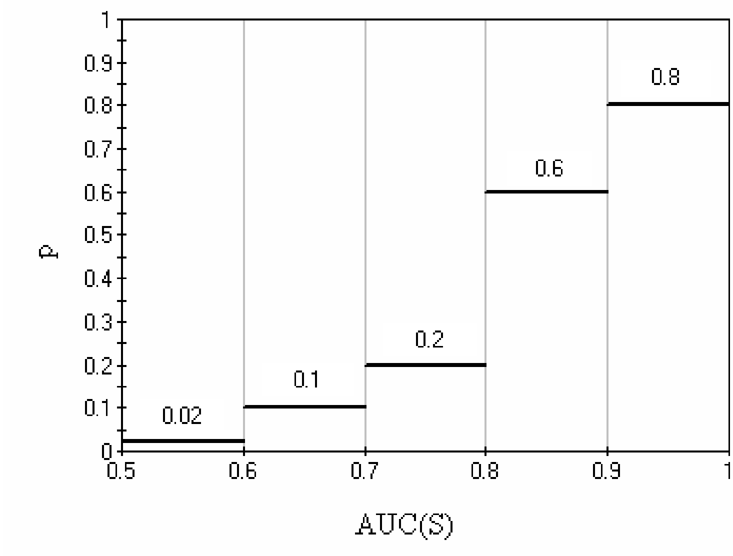


Рисунок 1 – Зависимость ρ от AUC_S

При ее построении учитывалась градация качества классификаторов в зависимости от значения площади под кривой [4] (табл. 1).

Таблица 1

Интервал AUC	Качество модели
0,9 – 1,0	Отличное
0,8 – 0,9	Очень хорошее
0,7 – 0,8	Хорошее
0,6 – 0,7	Удовлетворительное
0,5 – 0,6	Неудовлетворительное

Эксперименты на искусственных наборах данных

Целью проводимых экспериментов являлось сравнение эффективности функции приспособленности с фиксированным и переменным параметром ρ, а также оценка работы алгоритма при больших размерностях.

В качестве первого набора данных использовалось искусственно сгенерированное множество из функциональной зависимости $f(\mathbf{X}) = x_1 - x_2 - x_3 + x_4 x_5^2 \geq -0{,}7$, x_k ($k = 1,..,5$) равномерно распределен на (0;1). Объем обучающего и валидационного множества сос-

тавил 375 и 125 записей соответственно. К набору данных были добавлены 45 случайных переменных, также равномерно распределенные на (0;1). Таким образом, общая размерность пространства поиска составила 2^{50}.

Результаты экспериментов сведены в табл. 2. Для генетического алгоритма приведен результат с минимальным (*n*) и средним (n_{cp}) числом переменных на 100 запусках алгоритма с постоянным параметром $\rho_c = 0{,}02$ и изменяющимся в зависимости от значения *AUC* ($\rho_d \in [0{,}02;0{,}8]$).Число хромосом в популяции генетического алгоритма 30, алгоритм останавливался при постоянстве целевой функции в течение 20 эпох. Коэффициенты логистической регрессии рассчитывались стандартным методом Ньютона. В таблице также приведены результаты для процедур *forward* (*f*) и со всеми включенными переменными (*e*), *h* – число эпох генетического алгоритма. Процедура *backward* не включила в модель ни одну переменную (расчет пошаговой регрессии производился в пакете SPSS 14,0).

Таблица 2 – Результаты экспериментов для первого набора

Модель	AUC_S	*n*	n_{cp}	*h*
ГА (ρ_c)	0,963	8	8,7	64
ГА (ρ_d)	0,963	5	17,2	79
f	0,95	50	–	–
e	0,896	15	–	–

Целевая функция с переменным параметром ρ_d показала себя лучше всех: несколько раз генетический алгоритм находил решение именно с теми 5 переменными, на основе которых была получена выходная переменная.

В последние годы проблема отбора признаков (англ.: *feature selection*) в машинном обучении приобрела самостоятельное значение, и для тестирования алгоритмов был разработан ряд искусственных наборов данных сложной природы с большим числом шумовых признаков (с долей от 30 до 95 % от их общего числа). Задачи с таким числом признаков на практике возникают нечасто, но эти наборы данных служат для проверки масштабируемости алгоритмов. Из *UCI Machine Learning Repository* был взят набор данных *Madelon* [5], состоящий из 500 входных переменных, из которых значимыми (на основе которых генерировалось выходное поле) являются только 20. Остальные переменные были искусственно добавлены в набор, а их значения имели распределения, близкие к истинным переменным, что усложняет задачу отбора переменных. Обучающее множество *Madelon* имеет размер 2000 записей, валидационное – 1000.

В результате работы генетического алгоритма уже на 8 эпохе число переменных было сокращено до 224 с $AUC_S = 0{,}60$ (AUC_S на полной модели тоже равен 0,60). Поэтому всего за несколько эпох генетического алгоритма можно существенно снизить число переменных. Процедуре *forward*, к примеру, не удалось за приемлемое время выдать решение (пакет SPSS 14.0).

Применение в задаче кредитного скоринга

В качестве практической задачи были взяты реальные кредитные истории, содержащие информацию о качестве обслуживания долга заемщиками и их социально-экономические параметры: возраст, образование, количество лет проживания в регионе, доход и т.д. – всего 20 переменных. При помощи специального преобразования непрерывное

множество переменных, отвечающих за число просрочек, было трансформировано в бинарную переменную «плохой/хороший заемщик». Обучающее множество составило 3557 записей, валидационное – 687. Доля «плохих» заемщиков составила 17 %. Множества отличались тем, что в них присутствовали шумы, аномалии и незначащие факторы. Пока что такая картина является скорее нормой в кредитных историях российских банков, нежели отклонением.

Генетический алгоритм из 20 переменных составил только 2 с $AUC_s = 0{,}66$. Полная модель имела площадь под кривой 0,6.

Заключение

В целом генетический алгоритм с предложенной кусочно-линейной штрафной функцией за включение в модель новых переменных, зависящей от площади под кривой на текущей эпохе, показал хорошие результаты на синтетических наборах данных и в задаче кредитного скоринга, лучшие, чем при использовании фиксированного параметра штрафа с $\rho = 0{,}02$. Однако подход обладает рядом недостатков. Во-первых, он имеет высокую вычислительная сложность, которая выражается в том, что на каждой эпохе необходимо решать регрессионное уравнение и рассчитывать площадь под кривой. Поэтому для получения результатов за приемлемое время мы имеем ограничения на размер популяции. Во-вторых, метод Ньютона для расчета коэффициентов логистической регрессии иногда не сходится, и генетический алгоритм приходится запускать повторно. Поэтому можно сказать, что применение генетического алгоритма оправдано в задачах с пространством поиска, не превышающего 100 переменных. Кроме того, перспективным представляется сравнение подхода с другими эволюционными стратегиями, в частности, муравьиными алгоритмами.

Литература

1. Larose, Daniel T. Data Mining methods and models. – John Wiley & Sons, Inc., Hoboken, New Jersey, 2006. – 322 P.
2. Черкашенко Н.Ч. Этот загадочный скоринг // Банковские дело. – 2006. –№ 3. – С. 42-28.
3. Vinterbo S., Ohno-Machado L. A genetic algorithm to select variables in logistic regression: example in the domain of myocardial infarction // Journal of the American Medical Informatics Association. – 1999. – №6. – P. 984-988,
4. Zweig M.H., Campbell G. ROC Plots: A Fundamental Evaluation Tool in Clinical Medicine // Clinical Chemistry. – 1993 – Vol. 39, №. 4.
5. Madelon Data Set. – Режим доступа: http://archive.ics.uci.edu/ml/datasets/Madelon.

М.Б. Паклін

Відбір змінних в логістичну регресію генетичним алгоритмом

У статті досліджуються ефективні процедури відбору змінних в бінарні класифікуючі моделі на основі логістичної регресії. Для цього використовується генетичний алгоритм, причому у функцію фітнеса особини параметр штрафу за включення в модель нових змінних змінюється залежно від розрахованого значення площі під ROC-кривою. Проведені експерименти на модельних наборах даних і в задачі кредитного скорингу.

N.B. Paklin

Feature Selection in a Logistic Regression by Genetic Algorithm

In the paper we discuss effective procedures for a feature selection problem in a binary logistic regression model. A genetic algorithm was used to find best feature combinations, with the special fitness function based on a penalty parameter for including new variables. This parameter depends on ROC-curve index on current epoch. Experiments on Madelon data set and credit scoring classification problem were made.

Статья поступила в редакцию 21.07.2008.